NASA
Technical
Paper
2672

December 1986

Applications and Requirements for Real-Time Simulators in Ground-Test Facilities

Dale J. Arpasi and
Richard A. Blech

NASA

NASA
Technical
Paper
2672

1986

Applications and Requirements for Real-Time Simulators in Ground-Test Facilities

Dale J. Arpasi and
Richard A. Blech

Lewis Research Center
Cleveland, Ohio

National Aeronautics
and Space Administration

Scientific and Technical
Information Branch

Summary

The development and operation of a large, ground-test facility, such as a wind tunnel, can be accomplished more effectively and efficiently if computer simulations of the facility components and subsystems are available. The simulations can be used to predict the dynamic responses of the facility under normal and abnormal operating conditions before the facility becomes operational and can be used as test beds for designing and evaluating various facility control concepts. If the computer simulation can operate in real-time, it can be interfaced directly with the facility control system hardware and be used for operator training, prerun checkout of controls, and other applications where knowledge of the process dynamics is important. The use of such a simulator can result in reduced facility operating costs, improved productivity, and greater facility reliability and safety. This report discusses possible simulator applications and the associated simulator and control system design requirements. Several configuration options are discussed and a logical approach to configuration selection is presented. Procedures for determining specific simulator and control requirements for selected applications and configurations are given. Finally, some practical aspects of hardware selection are provided to aid in performing simulator cost-benefit analyses.

Introduction

The Lewis Research Center has been engaged in the development of an Altitude Wind Tunnel facility (AWT) to support research in low-speed aerodynamics and icing effects. As a part of this effort, a study was performed to determine the desirability of including a real-time simulator in the control and data systems of the facility. This simulator would be used as an alternative to, and in conjunction with, the actual operation of the facility to reduce operational costs and to improve reliability.

To determine the desirability of a real-time facility simulator, a cost-benefit analysis must be performed. For some of the intended applications, the inclusion of a simulator could impact the design of the facility's control and data systems; therefore, this cost-benefit analysis should be done early in the facility design. The costs of providing the simulator include the following: (1) initial procurement of the simulator hardware and software, (2) special interfacing to the control and data systems (as well as supplemental hardware and software interfacing costs imposed on these systems), and (3) ongoing maintenance and support costs for the simulator, including supplemental staffing requirements. Benefits are measured in terms of reduced operational costs and improvements in reliability and safety.

The purpose of this report is to provide the basis for a subjective cost-benefit analysis by objectively identifying potential applications and their benefits and requirements for implementation in terms of hardware, software, and support. The information presented is of a general nature and, therefore, should be applicable to many types of facilities. A general facility and control configuration is defined and used as the basis for discussion of the applications and requirements for a dedicated simulator. Simulator applications are discussed in general and then related to facility operation in terms of potential benefits.

The establishment of requirements begins with the identification of the simulator and control functions and data paths required to support the applications. The possible configurations for integrating a simulator into the facility are described and related to the benefits which they can provide. Finally, the specification of requirements for the facility design, pertinent to simulator integration, is discussed. A configuration selection procedure is provided to select the minimum cost configuration according to the desired benefits. The function and data path requirements are then related to the selection configuration. This is followed by a discussion of practical design and support considerations.

It is intended that the information presented will simplify the cost-benefit analysis by providing a comprehensive list of general applications and their requirements. With this information in conjunction with specific facility and control designs and operational procedures, specific simulator integration requirements may be established.

Facility and Controls

To apply the applications of real-time simulation to the operation of a wind tunnel facility and to establish simulator and interface requirements, it is necessary to establish relationships with the configuration of the facility and its control system. Unfortunately, in the case of new facilities, the facility design will not be established before the commitment must be made to incorporate a simulator. Since the purpose of this document is to provide general information for the selection of simulator applications and formulation of design specifications, a general facility and control system design will be used as the basis for discussions. It is intended that this "paper" facility will be sufficiently general, so that the actual facility will be a subset. If so, interpolation may be used to establish requirements and cost estimates for the selected applications.

The general facility and control system configuration selected for these discussions is functionally shown in figure 1. The facility consists of a tunnel (TUN), a number of subsystems (FS), and a test model (TM). The control system consists of subsystem controllers (FSC), under the supervision of the tunnel control (TC), and a test model controller (TMC). The control system interfaces to the facility through servos (FSSV, TMSV) and sensors (TS, FSS, TMS). It interfaces to the operator through the facility and test model control stations FCS, TMCS.

The tunnel contains a test section for mounting the test model, and the airflow across the model is provided and regulated by the subsystems. Tunnel operation is monitored through the tunnel sensors TMS by the operator and the TC. These sensed variables are termed SV, in figure 1.

Each subsystem and the test model has one or more components interfaced to the tunnel, and may have many more auxiliary components outside the tunnel. The operations of these components and, therefore, of the tunnel is controlled by servos. The servo outputs are the facility independent variables (IV). Subsystem and servo operation is monitored through the subsystem and test model sensors FSS and TMS. The FSC and TMC use sensor outputs SV for control feedback to provide servo setpoint commands.

The servo setpoint commands are the facility manipulated variables (MV). Each servo generally has a fail-safe override (FSO) which may be enabled directly by the operator (OC) in emergency situations. This override causes the subsystem or test model to go to a safe condition.

A more detailed diagram of the general control system functions is given in figure 2. Three levels of control are implemented through switches. (In this diagram and subsequent diagrams, switches are shown by diamond shapes and their position selection by circles.) The lowest level of control is direct regulation of the MV by the operator. This will be termed the primitive mode (PRIM). In this mode the operator sends setpoint commands PRIM SP directly to the servos and the control algorithms are merely tracking these commands. The second level of control is termed the manual control mode (MAN). In this mode the operator directly governs test model and subsystem operation by commands MAN SP from the control stations directly to the subsystem and test model control algorithms. The tunnel control algorithm is bypassed and tracks the manual setpoints. The highest level of control, the automatic mode (AUTO), allows direct control of the tunnel by the operator. Operator commands AUTO SP, via the FCS, are used to select test conditions. In this mode, all control algorithms are operating, the TC provides setpoints for the FSC, and the FSC provide the manipulated variables.

The control modes are selected by the operator. The

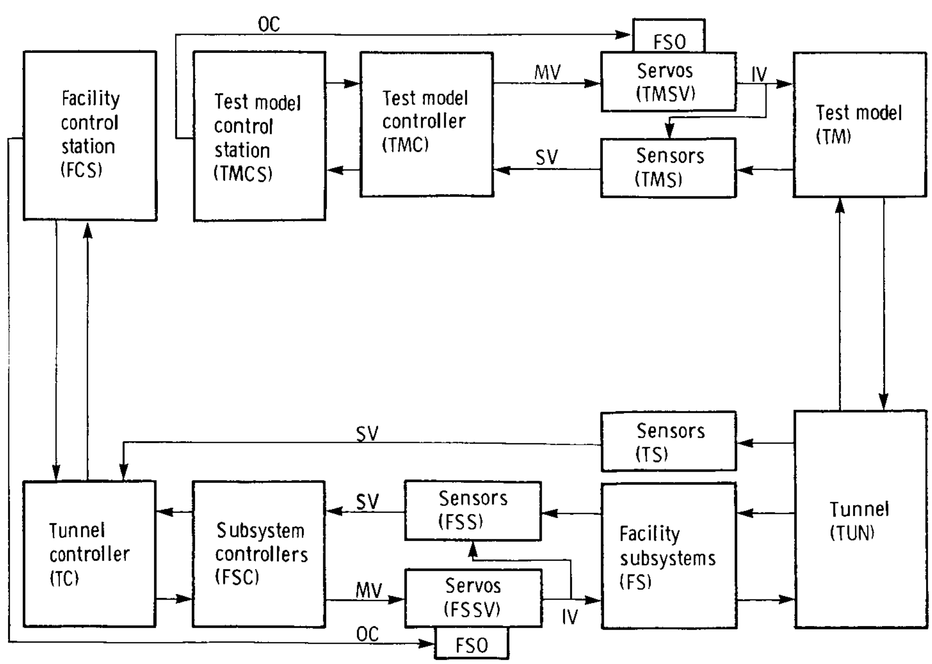


Figure 1.—General facility and control.

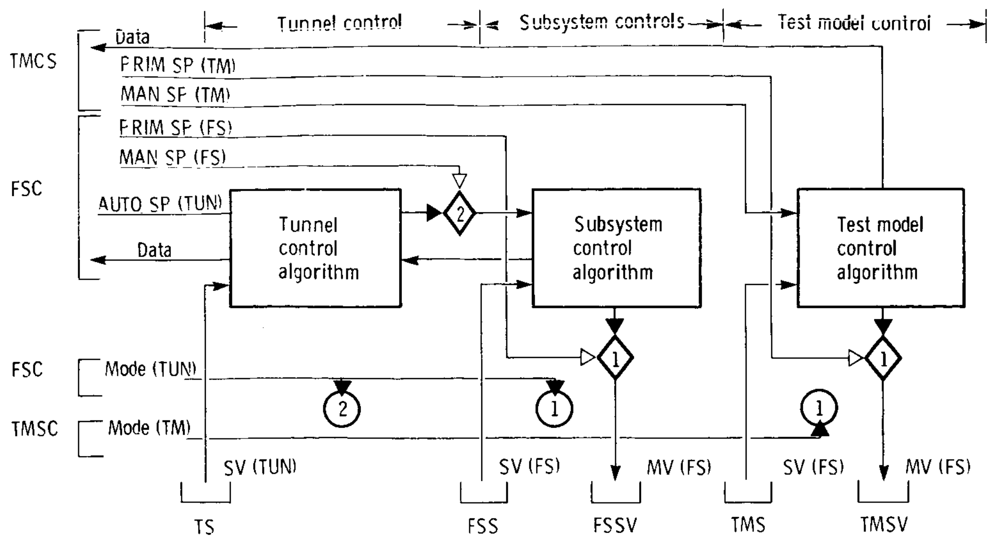


Figure 2.—Control system functions.

appropriate switch selections are shown in table I. Switch 1 is used to determine the source of the servo setpoints, and switch 2 is used to determine the source of the subsystem setpoints. Switch selections are denoted by the shading of arrowheads which correspond to the data path options shown in figure 2. (Solid arrowhead denotes switch is selected.) Three transition modes are also shown in table I. These may be used in facility startup and shutdown, operation under failures, and component testing.

Before concluding the description of the general facility and control system, it is necessary to describe an additional function which is assumed to be assigned to the operator control station. Besides setpoint command and mode selection capabilities the control stations provide all operational analysis necessary for operation of the facility. This purpose is to relate facility and model information to the operator by using displays. The analysis function, as well as the control function, may benefit from integrating a simulator into the control system.

Applications

The purpose of this section is to identify areas of application of real-time simulation to facility operation. The discussion begins with a definition of the various types of simulations

TABLE I.—CONTROL MODES

Control mode[a]	Subsystem[b]				Mode description	Comments
	i		*j*			
	S1	S2	S1	S2		
1	▲	▲	▲	▲	Automatic control	All control algorithms are operational; operator selects tunnel setpoint
2	▲	△	▲	△	Manual control	Tunnel optimal control is disabled; operator selects subsystem setpoints
3	△	△	△	△	Primative control	Tunnel and subsystem controls are disabled; operator selects servo setpoints
1.2	▲	▲	▲	△	Transition AUTO/MAN	Control modes may be used in staged startup and shutdown, failure mode operation, and off-line testing
1.3	▲	▲	△	△	Transition AUTO/PRIM	
2.3	▲	△	△	△	Transition MAN/PRIM	

[a]Modes 1, 1.2, and 1.3 are not available for test model subsystem operation.
[b]Switch selections are denoted by solid arrowheads which correspond to the data path options shown in figure 2.

which are pertinent. This is followed by a description of simulation applications in general. Finally, facility operational procedures are discussed and the potential applications of simulations to these procedures are defined. Each application is discussed in terms of potential cost savings and improvements in overall reliability and safety.

Simulation types

Five types of simulations will be used in the application discussions and are as follows:

(1) Nominal
(2) Operational
(3) Off Design
(4) Failure
(5) Predictive

Each is a representation of the facility. A nominal simulation is that representation of the facility used to design the facility control algorithm. It is initially derived by using mathematical modeling techniques supported by a limited amount of experimental data. An operational simulation is that representation of the facility which is used to verify control and facility operation and procedures. It generally begins as the equivalent to a nominal simulation, but evolves into a more or less empirical representation of the facility as experimental data becomes available. Similarly, a nominal simulation may evolve into its operational equivalent if the control design is reestablished when the simulation is modified to match further experimental data. The remaining simulation types are subsets of the operational simulation.

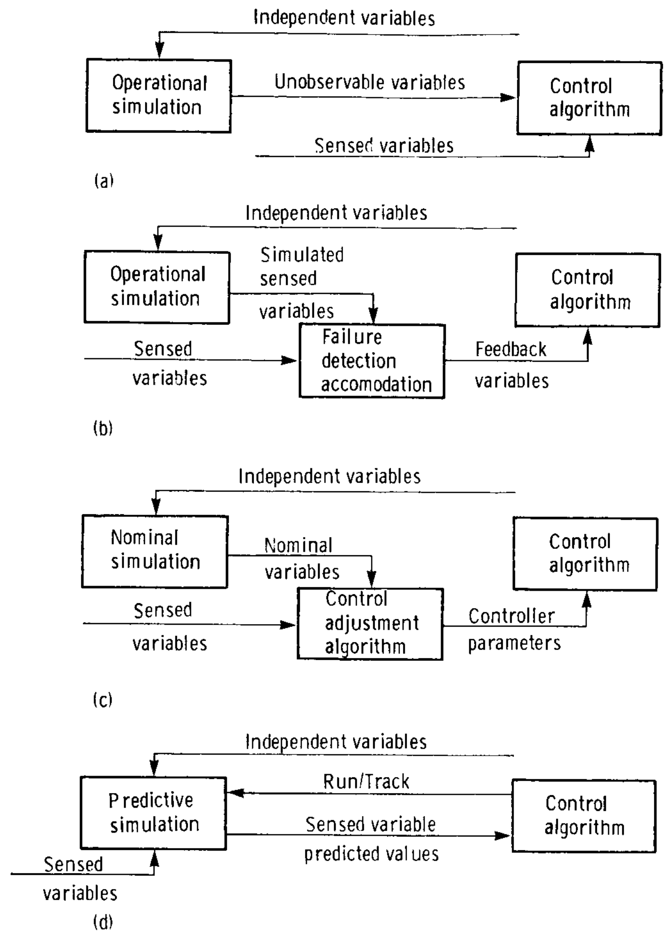
An off-design simulation is an intentional deviation from the operational simulation. Its primary uses are in control sensitivity studies and in simulating component degradation. This type of simulation is generally developed from an operational simulation by adjusting the simulation parameters to reflect potential variations in the facility components.

A failure simulation is used to simulate the failure of facility components for failure accommodation testing. It is developed from an operational simulation either by parametric adjustment or by passing simulation results through a failure synthesizer.

A predictive simulation is used to extrapolate simulation results to their values at some future time with simulation independent variables held constant at the time of measurement. It has application to both control optimization and to operational monitoring. A predictive simulation may be developed by running an operational simulation at an update interval faster than real-time in conjunction with a tracking mechanism which resets the simulation results to their sensed values at the real-time sampling interval.

General Applications

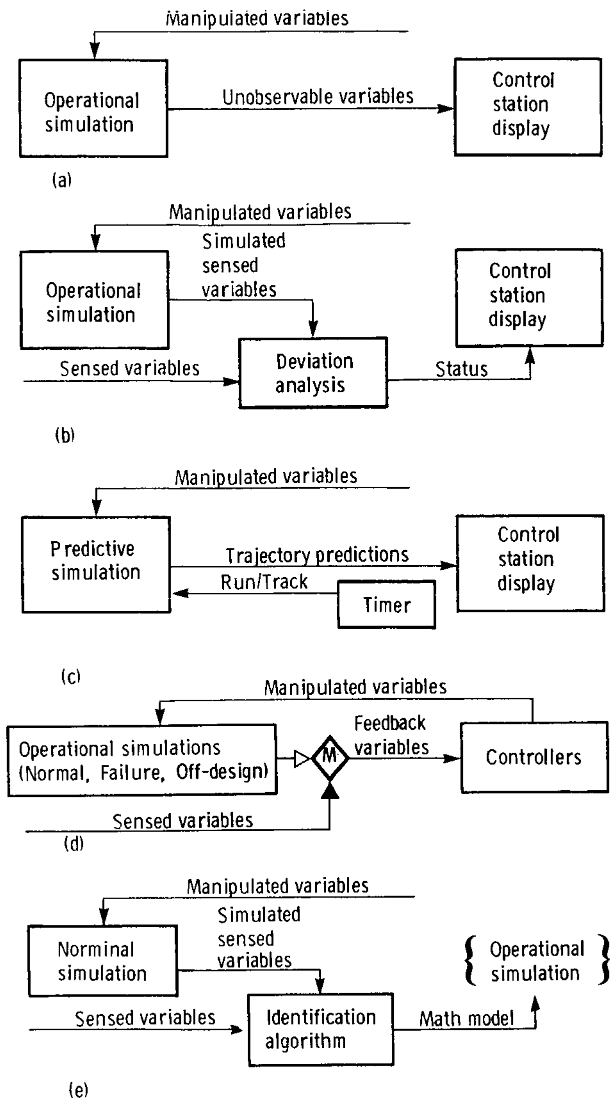
All of these simulations may be applied to improving control system performance and operational efficiency. Figures 3 and 4 depict nine general simulation applications. The first four (shown in fig. 3) are applications to improve control system performance. The last five (shown in fig. 4) are applications which may be used to improve operational efficiency of the facility. It should be noted that the control performance applications are included for the sake of completeness. The associated simulations are considered an integral part of the control design and should be distinct from those used for operational applications. They will not be included in the discussions of configurations and requirements, since they should be furnished as a part of the control system.



(a) Control of unobservable variables.
(b) Failure detections and accommodation.
(c) Adaptive control.
(d) Predictive control.

Figure 3.—General applications to facility control.

The use of an operational simulation to provide measurements of unobservable variables for control is shown in figure 3(a). The independent variables from the control algorithm and from external conditions are passed to the simulation. The simulation produces values of variables which are not sensed, but which are required for control purposes. These variables might be unobservable because of physical restrictions on the location of sensors, because sensing of these



(a) Generation of unobservable variables.
(b) Health monitoring.
(c) Trajectory predictions.
(d) Testing.
(e) Identification.

Figure 4.—General applications to facility operation.

variables is expensive, or because sensors with required dynamics are not available. A nominal simulation may be used if the control algorithm is sufficiently desensitized.

Figure 3(b) illustrates the application of an operational simulation to failure detection and accommodation. In this case, the simulation produces simulated sensed variables for comparison with actual sensed variables in the detection of failures. If a failure of a sensed variable is detected, the failure may be accommodated by using its simulated value for feedback into the control.

The application of simulation to adaptive control is shown in figure 3(c). The nominal simulation which was used to develop the design point control provides design point values of the sensed variables. These design point values are compared to the sensed variable values to provide a measure of the deviation of the system from nominal. This deviation is passed through an off-design control adjustment algorithm which modifies the parameters of the system control algorithm to optimize performance for the off-design condition.

Figure 3(d) illustrates the use of a predictive simulation to provide predictions of the value of a sensed variable at a future time. This look-ahead capability can improve control performance. The simulation operates in two modes. During the track mode, the simulation samples the sensed variables and adjusts its condition to reflect that of the system. During the run mode, the simulation is computed, starting from these initial conditions at a rate which provides values of the sensed variables at a future time. So, for example, if the control is being updated at a time ΔT seconds, then at every ΔT the simulation would enter the track mode and initialize its conditions according to the values of the sensed and independent variables. It would then enter the run mode and compute the simulation n times to produce predicted values at $n\Delta T$. The control algorithm would then be computed. If the track mode and the control algorithm required Tx seconds, then the simulation must be computed at an update rate of $(\Delta T - Tx)/n$ seconds to complete its task. This usually means that the simulation must run faster than real-time. However, since the real-time update interval for the simulation is usually much less than the update interval for the control, because of the internal dynamic considerations, real-time simulation operation may suffice if n is small enough.

Simulation applications to improve operational efficiency are shown in figure 4. The first application (fig. 4(a)) is the display of unobservable variables for the operator and for data recording. Using an operational simulation reduces the need for sensors to provide information only for monitoring purposes.

Figure 4(b) depicts the use of an operational simulation for health monitoring. This application is similar to the failure detection and accommodation application described above, but instead of directly affecting the control, deviations between simulated and sensed variables are monitored for operator action. The deviation analysis, in this case, provides information for failure detection by the operator and may be used to identify degradation in facility components.

The use of a predictive simulation to predict trajectories is shown in figure 4(c). This application would be most useful in the manual or primitive control modes to provide look-ahead capability to aid the operator in achieving and maintaining a facility setpoint through subsystem or servo manipulation. As these manipulated variables are adjusted, the predictive simulation, running faster than real-time, could project the steady state values of the sensed variables, which would be achieved if the manipulated variables were held at their current values. The simulation setup and operation for trajectory prediction is similar to that described for the predictive control application. Except, the run and track modes are controlled by a timer, set to provide sufficient simulation calculations to achieve steady state values. Also, the track values of the sensed variables are obtained from the first calculation in the previous run mode. It could be necessary to limit the rate of change of the manipulated variables to achieve useful results, and, probably would be necessary to couple this application with extensive human engineering to avoid operator confusion.

Figure 4(d) illustrates the use of simulations in the off-line testing of controls and operational procedures. In this application a simulation is used to provide sensed variable measurements instead of running the actual facility. The simulation may be an operational, off-design, or a failure simulation as required to meet test objectives. A switch is shown to provide selection between facility and simulation variables. This allows the control loops to be phased in and provides for staged start up of the facility.

The applications described above make extensive use of operational simulations. Their usefulness improves as the operational simulation becomes a closer representation of the facility. It is important to have a mechanism in place to improve the accuracy of an operational simulation. This mechanism, referred to as mathematical model identification, is depicted in figure 4(e). A nominal simulation is shown to provide simulated sensed variables. The values of these variables, obtained under controlled facility conditions, are compared to actually sensed variable values in an identification algorithm. On the basis of these comparisons, an artificial intelligence algorithm develops values of model parameters in order to improve the accuracy of the operational simulation. The nominal simulation is replaced by the operational simulation, and the process may be repeated until the accuracy of the resulting operational simulation is within the tolerance required by its applications.

To this point, simulation applications have been discussed in general terms. To assess the benefits of these applications, it is necessary to relate them to the operation of the wind tunnel. In the following discussion these relationships will be established along with a qualitative discussion of benefits as they relate to the general facility and control system. Quantitative benefits, in terms of cost savings and improvements in safety and reliability may then be derived from specific facility and control system design and operational specifications.

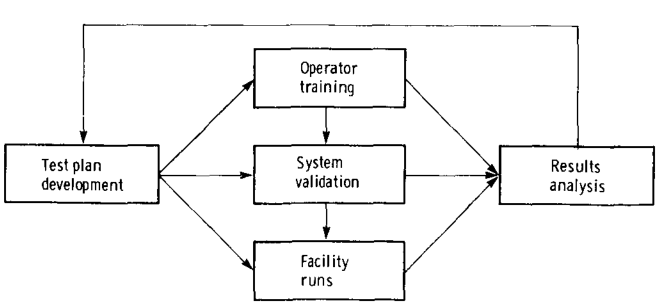


Figure 5.—Wind tunnel operational activities.

The activities associated with the operation of a wind tunnel are shown in figure 5. A facility is usually operated to test the effects of its operation on a test model or to verify the calibration and operation of the facility itself. The objective of operation is referred to as a test and the first step in facility operation is the development of a test plan. The test plan establishes procedures to be followed to achieve the objective. These procedures fall into one or more of the following activities: Operator training, facility and control validation, facility runs, and results analysis. The results analysis may be intermixed with the other activities and unsatisfactory results may require revisions of the test plan.

Applications to Test Plan Development

The development of a test plan involves a number of steps. It begins with a selection of tunnel operating conditions at which the model is to be tested. At each of these points a selection of model operating conditions is made. Following selection of these static conditions, dynamic requirements may be established for model and facility manipulations about and between these points. The next step would be to identify potentially hazardous operating conditions, which are to be avoided in moving from one condition to another. This would be followed by a failure effects analysis to assess probabilities of and costs associated with component failures. On the basis of the information established to this point, procedures are established for the other operational activities which will meet the required test objectives at a minimal cost with maximum safety. Simulations can be used to verify the effectiveness and efficiency of these procedures.

From a cost stand point, it is important to minimize facility operating time while insuring proper establishment of model test conditions. Using the testing application of figure 4(d) optimizes the sequence and procedures for establishing test conditions in order to minimize facility settling time at each condition.

The operation of a facility at the limits or outside of its design envelope may be necessary to meet model test requirements. Procedures for abnormal operation may be established and verified by using the testing configuration. Hazardous operating regions can be defined and avoidance procedures established, thereby improving the safety of the tests.

By simulating potential failures, the impact on the model and the facility can be estimated. Special overrides and operational procedures can be specified to minimize the impact of failures.

Once the test plan has been established, it may be validated by using the testing configuration. Using off-design simulations determines the sensitivity of the established procedures to variations between simulated and actual components. A procedure which causes problems in off-design operation is likely to cause problems during actual operation. Validating procedures by using simulations may provide improvements in both cost and safety.

Applications to operator training.—Proper training of operators is as important as optimizing control system performance. The operator must be able to operate the facility efficiently during both normal and abnormal operation. This includes tunnel and model manipulation, and requires a thorough understanding of the test plan procedures which relate to system validation, facility runs, and results analysis.

It is not cost effective or safe to train an operator by running the actual facility. This also holds for model operation training if the model is expensive and requires complex operational procedures. Using simulations to train operators produces both cost and safety benefits. Additionally, failure simulations may be used to familiarize the operator with the detection and handling of abnormal operation.

Applications to system validation.—System validation is important to facility control development, bringing the facility on-line, and prerun checkout of the control and data systems. During development, the control is optimized for both normal and abnormal operation. Sensitivity studies are used to insure proper operation for projected variations of facility components from nominal. When the facility and control system have been delivered, difficulties may be encountered in bringing them on-line. System validation procedures may be developed to minimize these problems and isolate malfunctioning components. When the facility is operational, an integral part of the operating procedure is the validation of the control and data system before each run.

Using simulations for testing control algorithms is standard practice. When the control system has been delivered, algorithm modifications and improvements are usually necessary, since the simulations used during development are only nominal representations of the control and facility. The testing application (fig. 4(d)) will permit quick assessment of control algorithm changes with actual control hardware. The sensitivity and controllability can also be validated by using off-design and failure simulations. Substantial improvements in reliability can be achieved by using actual control hardware and operational simulations, since the resulting software implementation of the improved algorithm may be switched directly for facility operation. This switching capability eliminates the need for reprogramming and eliminates the errors which may occur if a control simulation is used for validation. An operational simulation, resulting from actual test data, will improve the accuracy of design and validation.

Prerun checkout of the control and data systems is mandatory to insure safe and effective operation of the facility. This may be accomplished, to some extent, by using an open loop control (i.e., without sensed variable feedback). The control and data software can be tested by substituting dummy values for the sensors. The hardware can be tested by using diagnostic software. The open loop approach may not be sufficient, however, because these tests do not dynamically duplicate conditions during facility runs. Closed-loop validation can be incorporated as either an open-loop supplement or replacement by using the simulation testing application (fig. 4(d)). Critical test conditions can be established and run, which exercise all control paths. This approach to prerun checkout offers improvements in both reliability and safety.

The capability to stage the facility startup sequence is valuable in bringing the facility on-line during initial testing and during prerun checkout. This allows validation of facility components in a logical, sequential manner to avoid catastrophic failures. The first step in staged startup is the validation of the control system described above. Once this has been accomplished the facility subsystems must be started and brought under control. This may be done sequentially with the model subsystem first, the tunnel regulatory subsystems next, and the tunnel power-generation subsystems last. The individual sybsystems may be similarly started (under primitive control) by sequentially enabling their actuators and by gradually bringing the subsystem under manual control. Once all subsystems have been started, the facility can be brought gradually under automatic control.

Staged startup capabilities can be enhanced by using the simulation testing application (fig. 4(d)) and the applications to facility runs (to follow). The testing application allows the higher level of controls to be functioning while the actual subsystem is being controlled at a lower level. For example, a subsystem may be started under primitive control while the feedback values for the manual control are derived from the subsystem simulation. Subsystem performance may be evaluated by using the health monitoring application (fig. 4(b)). During startup, the subsystem's manual control output may be evaluated by the operator by comparison with the primitive setpoint commands. Once the subsystem has been started, the control output should duplicate these commands if the operational simulation is accurate. In this case, the subsystem can be manipulated under primitive control and the manual control output should statically and dynamically track the primitive commands. If they do not match, then either the operational simulation is in error, or a failure has occurred in the manual control. Manual control of the subsystem can be gradually incorporated by switching from simulated sensors to actual subsystem sensors. This can be done one at a time to isolate any problems with the control.

Using a simulator in conjunction with staged startup increases information and enhances procedural options. This leads to improvements in safety and reduction in the time and costs associated with this activity.

Applications to facility runs.—When running a facility, information which improves operator performance is important. Facility simulations may be used to improve the quality of this information. These applications are depicted in figures 4(a) to (d) and include health monitoring, trajectory predictions, and the display of the values of unobservable variables. The descriptions of these applications, provided in the subsection General Applications, relate directly to the running of the facility. The information produced enhances both cost and safety benefits.

Applications to results analysis.—A critical part of facility operation is the analysis of results obtained from facility runs. Simulations may be used to augment analytical procedures. The capability to produce simulated test results for comparison with actual results might be helpful in resolving uncertainties encountered during analysis. Fault isolation is another analytical application for a facility simulation. If a simulation can be used to speed up and improve the accuracy of postrun analysis, improvements in the operational efficiency of the facility will result through shorter turn around times and better selection of test points.

Summary of Applications

The applications of simulation to facility operation are summarized in table II. For each operational activity, the applicable general application is indicated in terms of figure 4 (e.g., the generation of unobservable variables corresponds to fig. 4(a)). The specific application to the activity is listed in terms of qualitative benefits. The primary qualitative benefits (cost, safety, and reliability), which are impacted by the application, are also listed.

Notice that the testing application (fig. 4(d)) is most used, and therefore, can be expected to provide the most benefits. The value of these benefits and those of the other applications must be based on a specific facility design and its operational complexity.

System Integration

The applications of simulators to wind tunnel operational activities require various degrees of integration with the control system. Some applications require integration within the actual control system, some may be achieved by integrating the simulator with duplicates of key control system components, and some merely require expansion of the simulator to include simulation of control system functions. The degree of integration directly impacts the cost of an application. It may be measured in terms of the complexity and accuracy of the simulation and the data transfer requirements necessary to support the application.

In this section, options and requirements for integrating a simulator into the facility control system will be discussed. The functions and data paths required to support all the simulator applications to facility operational activities will be established. Potential hardware configuration options for simulator integration are discussed in terms of control requirements and the applications which they support. Finally, the various simulations and supplemental capabilities required to support the configuration options are tabulated in terms of functional and data path requirements for reference in the section Requirements Specifications.

TABLE II.—SUMMARY OF APPLICATIONS AND BENEFITS

Activity	Simulation application[a]					Qualitative benefits	Quantitative impact (primary)
	Generation of unobservable variables	Health monitoring	Trajectory predictions	Testing	Identifi-cation		
Test plan development				●		Minimize operating points	Cost
	●			●		Identify hazardous regions	Safety
	●			●		Failure effects analysis	Safety
				●		Test plan validation	Cost/safety
Operator training				●		Off-line facility operation	Cost/safety
				●		Off-line model operation	Cost/safety
				●		Failure synthesis	Safety
System validation	●			●		Control algorithm testing	Reliability
				●		Actual control testing	Reliability
				●		Prerun checkout	Reliability
	●	●	●	●		Staged startup	Safety
Facility runs		●				Health monitoring	Safety
			●			Trajectory predictions	Cost
	●					Unobservable variables	Safety
Results analysis	●			●		Result interpolation	Cost
					●	Model identification	Cost
				●		Fault reproduction	Cost

[a]See figure 4.

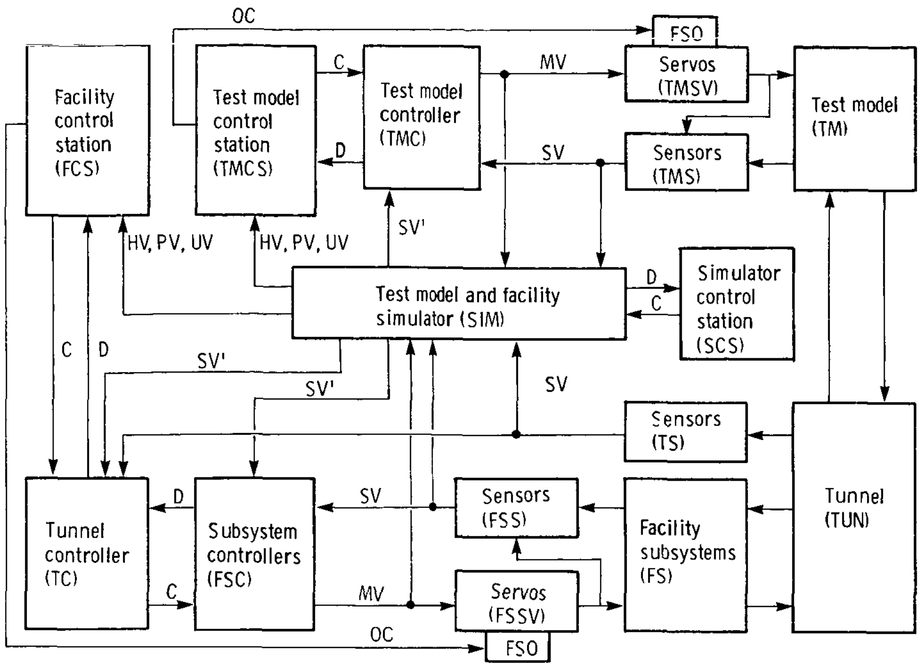


Figure 6.—General facility and control with close coupled simulator.

The general facility and control system, modified to include an all purpose simulation, is shown in figure 6. The simulation is supported by a simulator control station which provides commands and receives data to regulate and record simulator operation. Interconnections between the simulator and the controllers link the manipulated variables MV and the sensed variables SV to the simulator and the simulated sensed variables SV′ to the controllers. Connections from the simulator to the facility and test model control stations provide simulated variables for health monitoring (HV), trajectory predictions (PV), and for the display of unobservable variables (UV).

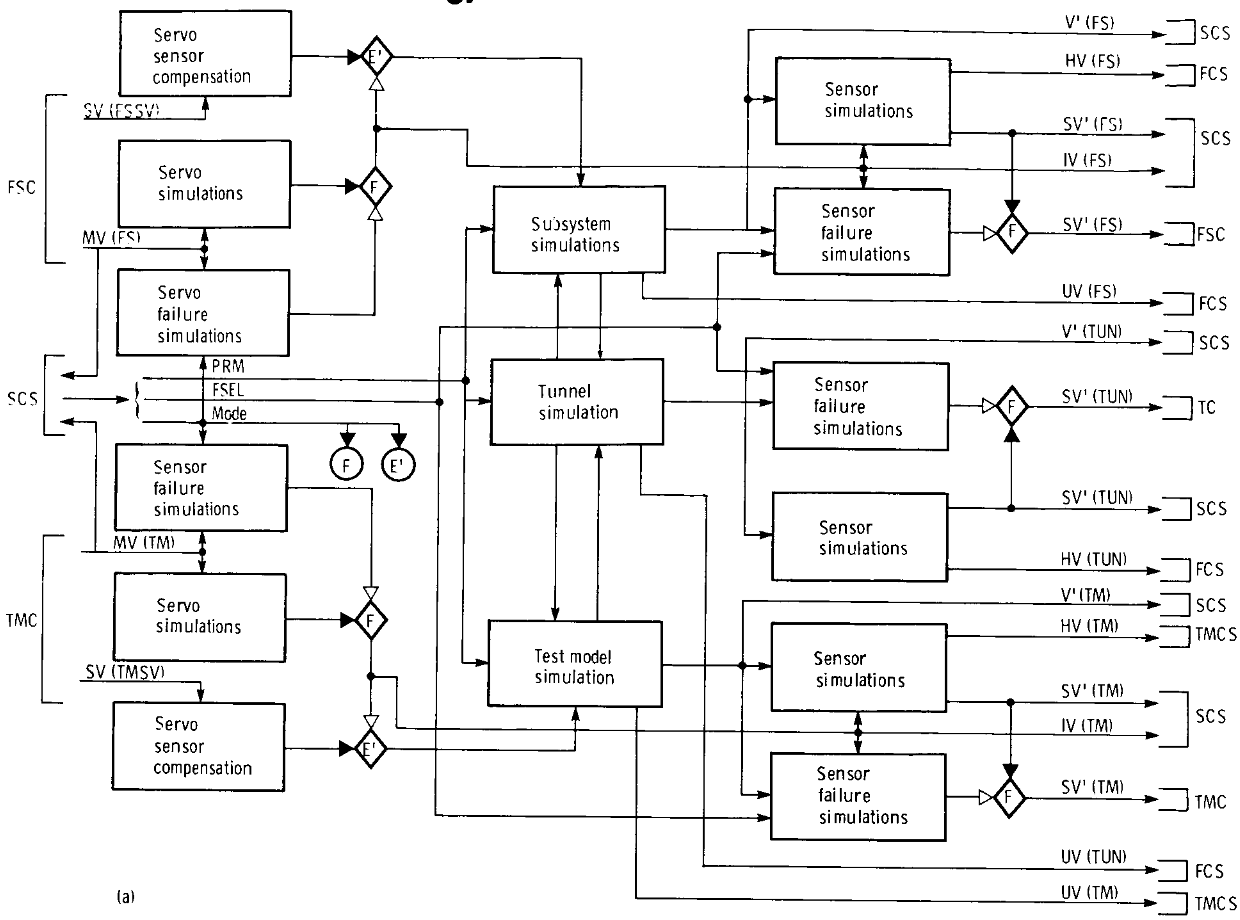
Simulator Functions

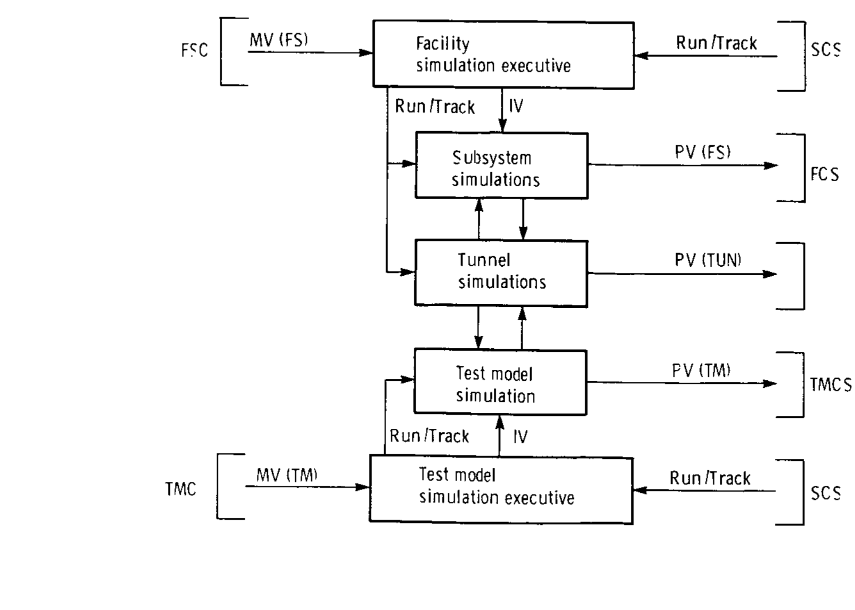
A functional diagram of the simulation is given in figure 7. The operational simulation is shown in figure 7(a). This simulation consists of simulations of the subsystems, tunnel, and the test model, supported by simulations of the facility servos and sensors. These elements are basic to all operational applications and are all that are required to support the generation of unobservable variables, health monitoring, and mathematical model identification (simulation applications shown in parts (a), (b), and (e) of fig. 4). The servo simulations are driven by the manipulated variables from the subsystem and test model controllers MV. The servo outputs (simulation independent variables IV) drive the subsystem and test model simulations, which are integrated into a complete facility simulation through the tunnel simulation. The unobservable variables UV are sent to the facility and test model control stations, as are the simulated variables, which are used for health monitoring HV after they are passed through appropriate sensor simulations. The math model identification application requires that the independent variables IV and the pre-sensed (V′) and sensed SV′ simulated variables be sent to the identification algorithm. It is assumed that this algorithm will reside in the simulator control station (SCS).

The testing application (fig. 4(d)) may require more extensive simulator capabilities. To support this application, servo and sensor failure simulators are included for failure synthesis. These are enabled by the F switches (fig. 7) via the mode command from the SCS. More than one failure may be simulated for each servo and sensor. If so, the particular failure is selected via the FSEL command. The operational simulation may be converted to an off-design simulation in order to support control algorithm testing by adjusting its parameters via the PAM command from the SCS. The simulated sensed variables (SC′) are then passed to the controllers from the enhanced simulation as replacements for the actual sensors for testing purposes.

One further enhancement to the operational simulation is necessary to allow the simulator to be used in the staged startup application. The sensed values of the servo positions are sent to the simulator to permit the simulation to respond to actual facility independent variables. These are obtained by passing the sensed values through a servo-sensor compensator which removes sensor dynamics. This feature is enabled by switch E′ in figure 7. This switch is shown as being controlled by the mode command from the SCS. It may be more convenient to control it from the appropriate facility controllers.

Figure 7(b) shows the simulator functions required for the trajectory prediction application (fig. 4(c)). The subsystem, tunnel, and test model simulations must be capable of being





(a) Operational simulation.

(b) Predictive simulation.

Figure 7.—Simulator functions.

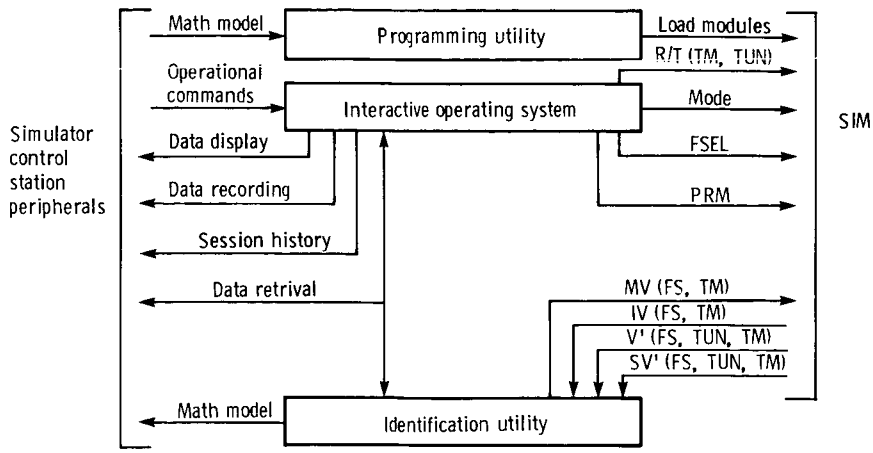


Figure 8.—Simulator control station functions.

calculated many times within a control update interval so that near-steady-state conditions are achieved for each calculation of the manipulated variables. This implies that if other simulation applications are run in parallel with this application, this simulation must be distinct from that of figure 7(a). The manipulated variables MV (FS and TM) are input from the subsystem and test model controls. These are processed by simulation executives which provide the independent variable IV to the simulation by inserting the servo dynamics. The executives also sequence the calculation of the predictive simulations according to the R/T = TUN, TM information sent from the SCS. To minimize the amount of data required, the first calculation in each predictive sequence is used to initialize the simulation for the next calculation sequence (as opposed to using actual sensed variables to provide these initial conditions). This implies that these simulations must have good static and dynamic accuracy. The predicted variables PV = FS, RUN, TM are sent to the facility and test model control station where predictive trajectories are formed and displayed to aid in operator setpoint selection.

Simulator control station functions.—The functional requirements of the simulator control station are shown in figure 8. The programming utility is used to generate the simulator program according to math model specifications from the operator. Initially, the math model is derived analytically. When facility data becomes available, the math model may be revised by using the identification utility and the simulator reprogrammed for improved accuracy in the operational and predictive simulations.

The identification utility accepts a time sequence of data recorded during a facility run which contains values of the manipulated and sensed variables necessary for identification of the tunnel, subsystem, or test model. The manipulated variables, MV (FS and TM) are sent to the simulator as driving functions. The simulator returns values of IV, V′, and SV′ at corresponding times. The utility develops a revised math model for the simulation elements by comparing the actual sensed values to the simulated variables and relating them to appropriate independent variables. The usefulness of this application, of course, depends on the capabilities of the identification utility.

The simulator control station must also contain an interactive operating system. It should have the capability to display, record, and retrieve simulation data, and it should maintain a session history to relate this data to operational activities. It must have interactive capabilities to enable monitoring and control of the simulation during operation. It provides the R/T, MODE, FSEL, and PRM commands to the simulator. As an example of interactive operation, consider the operator training application. While the operator is running the simulated facility through the FCS, an instructor may select failure simulations and induce them via FSEL and MODE in order to test operator reactions to these failures. The interactive capabilities may also be required to support model identification, trajectory prediction, and control testing.

Supplemental controller functions.—If the simulator is to be fully integrated with the facility control system, so as to permit simulated sensor substitution (required for the staged startup application), then certain features must be added to the facility controllers. These are shown in figure 9. This figure is an extension of figure 2. Note that a number of M switches have been added to enable the selection of either SV or SV′ as control feedbacks. Also, E switches have been added to enable or disable the transmission of the MV's to the servos. Each sensor may have an M switch, and each servo may have an E switch, or they may be switched in groups, depending on facility and test model characteristics and application requirements. It is recommended that these switches be incorporated into the control software to minimize cost and improve reliability. By proper switch selection through the FCS and TMCS, the operator can phase in actual hardware and phase out simulated hardware to gradually bring the facility on line and isolate problems with the controls.

The supplemental data paths (fig. 9) between the controllers and the operational simulator are described in the discussion of Simulator Functions (fig. 7). Note that the mode commands from the FCS and TMCS have been expanded to include regulation of the additional switches. The E′ mode command

TABLE III.—SUBSYSTEM OPERATING MODES UNDER THE MANUAL CONTROL MODE

Subsystem mode	Sensor M[a]		Servo E[a]		Mode description	Comments
	i	*j*	*i*	*j*		
1	▲	▲	▲	▲	On-line manual	Subsystem operating; control feedback from sensors
2	△	△	▲	▲	On-line primative	Subsystem operating; control feedback from simulation
3	△	△	△	△	Off-line	Subsystem not operating; full simulation
1.2	▲	△	▲	▲	Transition MAN/PRIM	Mode may be used in staged startup of subsystem and failure mode operation
2.3	△	△	▲	△	Transition PRIM/Off-line	

[a]Switch selections are denoted by solid arrowheads.

TABLE IV.—FACILITY OPERATING MODES

Facility mode[a]	Subsystem operator mode		Sensor M[b]		Mode description	Comments
	i	*j*	*i*	*j*		
1	1	1	▲	▲	On-line automatic	Facility operating; tunnel control feedback from sensors
2	1	1	△	△	On-line manual	Facility operating; tunnel control feedback from simulation
3	2	2	△	△	On-line	Facility operating; tunnel and subsystem control feedbacks from simulation
4	3	3	△	△	Off-line	Facililty not operating
1.2	1	1	▲	△	Transition AUTO/MAN	May be used in staged startup and failure mode operation
2.2	1	2	△	△	Transition MAN/PRIM	
3.2	2	3	△	△	Transition PRIM/Off-line	

[a]Modes 1, 1.2, and 1.3 not available for test model subsystem operation.
[b]Switch selections are denoted by solid arrowheads.

sent to the simulator is a reproduction of the E switch setting and may be used as an alternative to E′ selection from the SCS. This alternative permits automatic bypassing of the servo simulation when an actual servo is enabled.

The additional capabilities available for facility operation, which result from this integration of the simulator with the controls, are summarized in tables III and IV. In table III, the operating modes for the individual subsystems and the test model are shown. It is assumed that the subsystem controller is operating in the manual mode (control mode 2 in table I) and the operator is issuing manual setpoint commands. For illustration purposes, the subsystem is assumed to contain two sensors and two servos, *i* and *j*. Subsystem mode 3 shows all switches open so that the control feedbacks are simulated and the servos inoperative. Subsystem mode 2 shows the servos enabled with servo setpoints derived from the control. Since the control feedbacks are simulated, the control is essentially being used as a servo setpoint computer which accepts subsystem setpoints and translates them into servo setpoints. This is equivalent to the primitive control mode of table I with a very intelligent operator issuing servo commands. Subsystem operating mode 2 may therefore permit the elimination of

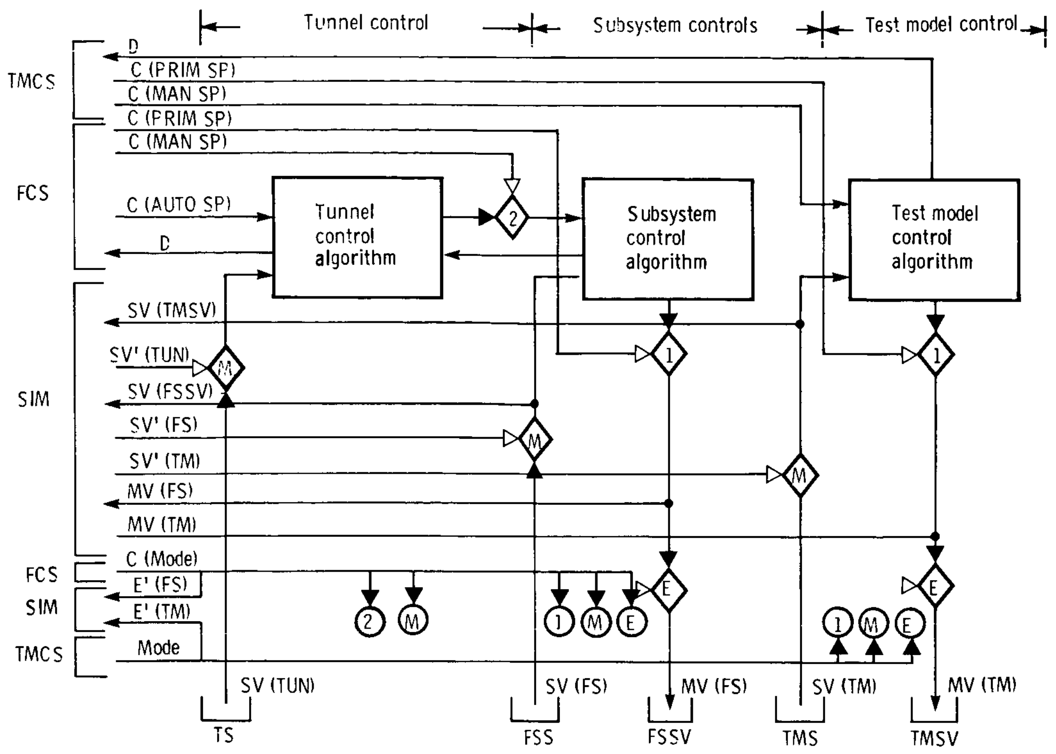


Figure 9.—Controller functions.

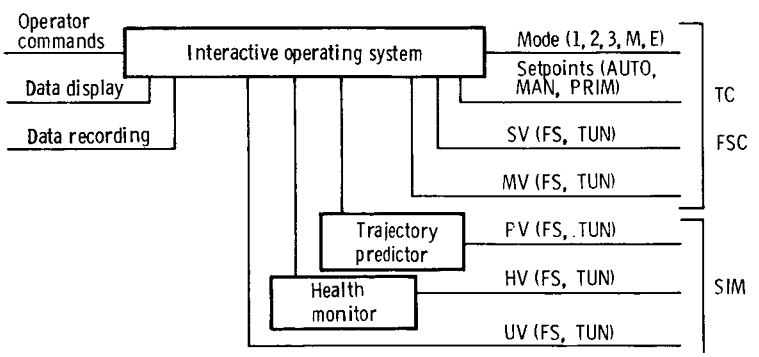


Figure 10.—Facility control station functions.

control mode switch 1. Subsystem mode 3 has all sensors and servos enabled and is the normal control mode for the subsystem. The transition modes 1,2 and 2,3 may be used for staged startup of the subsystem and for operation under sensor failures.

Table IV shows the facility operating modes resulting from this level of simulator and control integration. The control is assumed to be automatic (mode 1 in table I). The facility is assumed to contain two subsystems and two tunnel sensors. The facility operating modes result from various combinations of subsystem operating modes (table IV) and tunnel sensor switch positions. Facility operating mode 4 has the facility fully simulated. Mode 3 has all control feedbacks simulated, but with the subsystems operating in the primitive mode. In this case, the tunnel and subsystem controller are translating tunnel setpoint commands into primitive servo commands. The controllers are being used as calculators. Facility operating mode 2 has the subsystems fully on line in the manual control mode with the tunnel sensors simulated. In this case, the tunnel control is calculating the manual setpoint commands. Facility mode 1 has the facility operating under fully automatic control with the simulator fully off-line. Again, the transition modes shown in table IV may be used for staged startup and sensor failure accommodations.

Supplemental facility control station functions.—The functions of the facility control station required to support the simulator applications are shown in figure 10. The functions of the TMCS are similar. The control station must contain an interactive operating system as the interface between the operator and facility operation. This system should contain certain features to support simulator applications. In many cases, these features will already exist as support for other operational capabilities.

The mode command must be expanded to support the M and E switches in the control if their applications are selected. Sensed and manipulated variables, as required for the identification, must be sampled and recorded at rates required to support this application. Special primitive, manual, or automatic setpoint manipulations might also be required to support this application. Special processing algorithms may be required to support and enhance the predictive trajectory and health monitoring applications. In any case, a data transfer interface to the simulator is required for HV, PV, and UV if these applications are selected.

Configuration

The functions and data paths required to support the various simulator applications have been described. To this point, however, no consideration has been given to the hardware and software complexity necessary to meet these requirements. It is important to the cost-benefit analysis to minimize this complexity for the applications selected. In this section, a number of configuration options will be considered for integrating a simulator into facility operation. Each option will be discussed in terms of the operational benefits it provides and the complexity of the hardware and software required for its support. From this information, the minimum configuration in terms of complexity and, therefore, cost may be selected for a desired set of applications.

Figure 11 depicts the various configurations available for simulator integration. The simulator (SIM) along with its control station SCS is the central configuration element, and the configuration options represent variations in the connection of the simulator to the facility control system. These connections or data paths are shown as hexagons in the figure. To describe all of the options available, both the actual control system and an emulated (EM) control system are included. Each system contains the elements of the general control system (fig. 1) and duplicate hardware and software. Only the actual control system is shown connected to the facility servos and sensors. Data path A connects the simulator to the actual tunnel and facility subsystem controllers (SIM-TC, SIM-FSG), and A′ connects it to the test model controller (SIM-TMC). Data path A1 connects the simulator to the facility control station (SIM-FCS) and A1′ connects it to the test model control station (SIM-TMCS). The B, B′, B1, and B1′ data paths provide similar connections to the emulated control system. The dashed lines between the FCS and the FCS(EM) and between the TMCS and TMCS(EM) are included to show alternatives to the A1 and A1′ paths when the emulated control system is included in the facility.

The configuration options available, by including or excluding data paths, are shown in table V. The test model data paths are excluded from the table for the sake of simplification. Each configuration is numbered for future reference. Of the 16 possible configurations, only configurations 0, 1, 3, 4, 7, 12, and 15 will be considered as valid options for discussion. The others may be valid to satisfy particular facility needs, but offer no particular advantage from a simulator integration standpoint.

Configuration 0 provides for no connection between the simulation and the control system and, therefore, places no requirements on control system design. This stand-alone simulator would contain the facility and test model simulations and also nominal simulations of the control hardware and software. Since it is not connected to the facility control system, it need not be dedicated to the facility or be required to function in real time. It may be used to support the test plan

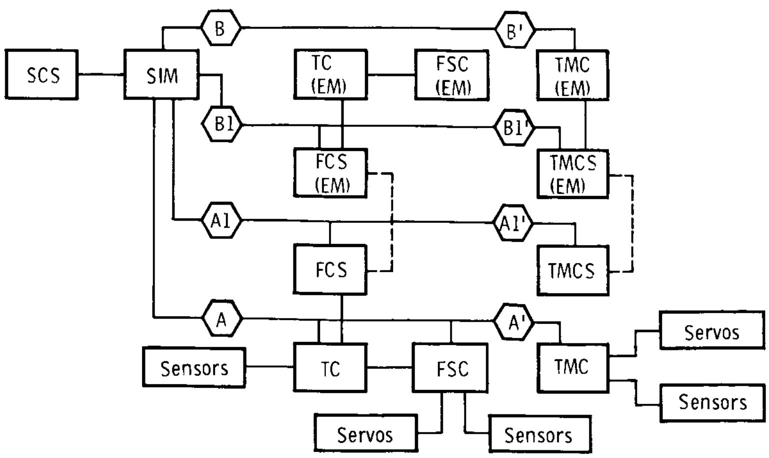


Figure 11.—Simulator integration options.

TABLE V.—CONFIGURATION OPTIONS

Configuration number	Data paths used				Controller hardware	Simulator dedicated	Integration
	A	A1	B	B1			
0	---	---	---	---	Simulated	No	None
1	---	---	---	Yes	Simulated	Yes	None
2	---	---	Yes	---	--------	---	--------
3	---	---	Yes	Yes	Emulated	Yes	None
4	---	Yes	---	---	Simulated	Yes	FCS only
5	---		---	Yes	--------	---	--------
6	---		Yes	---	--------	---	--------
7	---	↓	Yes	Yes	Emulated	Yes	FCS only
8	Yes	---	---	---	--------	---	--------
9		---	---	Yes	--------	---	--------
10		---	Yes	---	--------	---	--------
11		---	Yes	Yes	--------	---	--------
12		Yes	---	---	Actual	Yes	Total
13			---	Yes	--------	---	--------
14			Yes	---	--------	---	--------
15	↓	↓	Yes	Yes	Dual	Yes	Total

development activity, to test control algorithm changes during system validation, and to provide a means for result interpolation following facility runs.

Configuration 1 is similar to configuration 0 but includes an emulated facility and/or test model control station. The simulation requirements must be expanded to include all information transmitted between the actual FCS and the actual controllers. This configuration extends the potential applications to include operator training and mathematical model identification. The identification application requires a control station interface to allow transmission of test results to the identification algorithm in the SCS. If other result transfer mechanisms are incorporated, then this application may be supported by configuration 0. Configuration 1 also places no requirements on facility control system design.

Configuration 3 includes a complete emulation of the facility control system. It contains no connections to the actual control system and, therefore, does not impact facility control design. In this case, the simulator need contain only operational simulations of the facility, since actual control hardware and software are used. With this configuration, the potential applications are extended to include off-line testing of actual control hardware and software, and off-line reproduction of faults encountered during facility operation. Selection of configuration 3 requires incurring the cost of the emulated control system. This cost must be offset by the benefits incurred from the off-line testing applications or by non-simulator-related applications, such as the requirement for backup control hardware.

Configuration 4 is identical to configuration 1, except the actual control stations interface to the simulator. The controllers are again simulated and the off-line testing benefits of configuration 3 are not available. This configuration offers the benefits associated with the facility run activity. The generation of unobservable variables, health monitoring, and trajectory prediction applications are supported. The control design is impacted because of the A1 data path and the supplemental software functions required in the control stations to implement these applications. (See fig. 10.)

Configuration 7 is a combination of configurations 3 and 4. It provides all benefits except those related to system validation, which require simulator interfacing to the actual control hardware. It requires an emulated control system and impacts the control design as indicated for configuration 4.

Configurations 12 and 15 fully integrate the simulator with the actual control system. All benefits are available with these configurations. The impact on control system design is also maximized. In these configurations, the control feedbacks may be switched between the actual and simulated sensors in order to provide benefits to the on-line system validation activities of staged startup and prerun checkout. To provide these benefits, the controls must provide the switching capabilities, and the control stations must provide the switch selection capabilities. The servo simulations must be expanded to include servo-sensor compensation, thus allowing the servo outputs to be used as the independent variables of the facility simulation. If the facility run applications are required in this configuration, then appropriate data processing functions as described for configuration 4 must be included in the control stations.

Configuration 15 is identical to configuration 12, but with the emulated control included. This has the advantage of allowing concurrent facility operation. For example, control studies can be performed on the simulated system during facility runs.

Table VI summarizes the capabilities of the various options to support the simulation applications. Those applications relating to the test activity, for example, are supported by all configurations. Those related to on-line system validation are only supported by configurations 12 and 15. The impact on

TABLE VI.—CONFIGURATION SUPPORT FOR SIMULATOR APPLICATIONS

Simulation application		Supportive configurations used						
Activity	Benefit	0	1	3	4	7	12	15
Test plan development	Minimize operating points	Yes	Yes	Yes	Yes	Yes	Yes	Yes
	Identify hazards	↓	↓	↓	↓	↓	↓	↓
	Failure effects	↓	↓	↓	↓	↓	↓	↓
	Plan validation	↓	↓	↓	↓	↓	↓	↓
Operator training	Off-line operation	---	Yes	Yes	Yes	Yes	Yes	Yes
	Failure synthesis	---	Yes	Yes	Yes	Yes	Yes	Yes
System validation	Control algorithm testing	Yes	Yes	Yes	Yes	Yes	Yes	Yes
	Actual control testing	---	---	Yes	---	Yes	↓	↓
	Prerun checkout	---	---	---	---	---	↓	↓
	Staged startup	---	---	---	---	---	↓	↓
Run-time support	Health monitoring	---	---	---	Yes	Yes	Yes	Yes
	Trajectory predictions	---	---	---	Yes	Yes	Yes	Yes
	Unobservable variables	---	---	---	Yes	Yes	Yes	Yes
Results analysis	Result interpolation	Yes	Yes	Yes	Yes	Yes	Yes	Yes
	Model identification	---	Yes	Yes	Yes	Yes	Yes	Yes
	Fault reproduction	---	---	Yes	---	Yes	Yes	Yes

TABLE VII.—SIMULATION SELECTION MENU

Simulation required	Data paths required					Simulator function required
	SIM-SCS	SIM-TC/FSC	SIM-FCS	SIM-TMC	SIM-TMCS	
Tunnel and subsystem operational (TUN-OPR)	PRM(TUN) FSEL(TUN) Mode F(TUN)	MV(FS) SV′(FS,TUN)				TUN,FS,FSSV,FSS,TS, FAIL(FSS,V,FSS,TS), Switch F
Test model operational (TM-OPR)	PRM(TM) FSEL(TM) Mode F(TM)			MV(TM) SV′(TM)		TM,TMSV,TMS, FAIL(TMSV,TMS), Switch F
TUN controls operational (TC/FSC-OPR)			Setpoints Mode 1,2 MV(FS) SV′(FS,TUN) FSO(FSSV)			TC,FSC
TM controls operational (TMC-OPR)					Setpoints Mode-1 MV(TM) SV′(TM) FSO(TMSV)	TMC
Tunnel predictive (TUN-PRD)	R/T(TUN)	MV(FS)	PV(FS,TUN)			Predictive (TUN,FS)
Test model predictive (TM-PRD)	R/T(TM)	MV(TM)	PV(TM)			Predictive (TM)

the control design and, therefore, the integration complexity increases proportional to configuration number. Complexity and therefore cost may be minimized by selecting the minimum configuration number which can support the desired applications. There are, of course, qualifications necessary to this statement. If the cost of any required, emulated hardware cannot be absorbed by other facility applications (other than related to the simulator), then they must be added to simulator cost considerations. On the other hand, if a more complex configuration would promote operational efficiency, through convenience, or if expansion of simulator applications at a later date is desirable, then these considerations should be considered additional benefits. An approach to configuration selection is discussed in the section Requirements Specification.

Summary of Integration Requirements

Table VII is a simulation selection menu. The simulations required to support all of the applications and configuration options are included. The data transfer requirements for each simulation are identified, as are the simulator functions required. The parenthetical name assigned to each simulation will be used for reference in the section Requirements Specification.

Table VIII is an application selection menu. It lists the special data paths and functions required to support the general applications of figure 4. Again, nomenclature is established for reference.

Requirements Specification

This section establishes procedures and considerations used to define the design requirements for integrating a simulator into a facility with operational activities. The first consideration is to review the potential applications and assign a value for cost, safety, and reliability improvements in facility operation to each application. Next, the cost associated with each selected application must be defined. Finally, those applications which provide cost savings are selected, the configuration is specified, and the requirements are defined.

The section Applications provides general considerations for the evaluation of benefits. The section System Integration discusses function, data path requirements, and configuration options related to those applications. In this section, a structured approach to requirement specification, based on application selection, is presented. It may be used to iteratively perform a cost analysis for various applications and to develop

TABLE VIII.—APPLICATION SELECTION MENU

Application required	Supplemental data paths required (path/data)	Supplemental functions required (component/function)
Unobservable variables (Supplement A)	SIM-FCS/UV(FS,TUN) SIM-TMCS/UV(TM)	--------------------- ---------------------
Health monitoring (Supplement B)	SIM-FCS/HV(FS,TUN SIM-TMCS/HV(TM)	FCS/Health monitor TMCS/Health monitor
Trajectory prediction (Supplement C)		FCS/Trajectory predictor TMCS/Trajectory predictor
Sensor substitution (Supplement D)	FCS-TC/Modes M,E FCS-FSC/Modes M,E SCS-SIM/Mode E′ TMCS-TMC/Modes M,E	TC/Switches M,E FSC/Switches M,E FCS/Mode M,E SIM/Sensor compensation SIM/Switch E′ TMC/Switches M,E FCS/Modes M,E SIM/Sensor compensation SIM/Switch E′
Math model identification (Supplement E)	SIM-SCS/V′(FS,TUN,TM) IV(FS,TUN,TM) SV′(FS,TUN,TM) FCS-SCS/MV(FS) SV(FS,TUN) TMCS-SCS/MV(TM) SV(TM) SCS-SIM/MV(FS,TM)	SCS/Identification utility

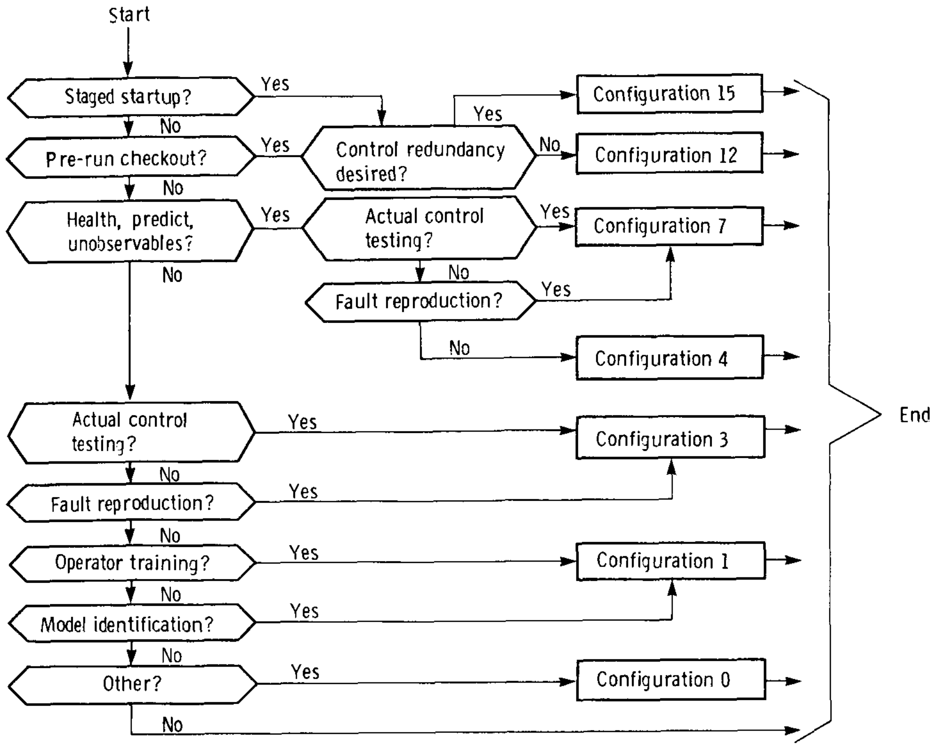


Figure 12.—Configuration selection.

a final set of requirements. Also included in this section, is a discussion of component selection, data transmission, and software support as they impact the practical aspects of simulator integration. An example cost analysis is provided for a simple facility.

Configuration Selection

Figure 12 illustrates a logical approach to configuration selection based on required applications. As shown in table IV, a configuration supports a group of applications. This suggests that there is a hierarchy of application considerations which dictates the configuration required. The first applications to be considered are those related to integration of the simulator with the actual controls. These are staged startup and prerun checkout. If these applications are desired, then either configuration 12 or 15 must be used. If control redundancy is desired for operational convenience or other considerations, then configuration 15 should be used. If either of these configurations are selected, then the data paths exist for all the other applications.

If the staged startup or prerun checkout applications are not desired, then the selection diagram proceeds down the order of applications. The next applications considered are those which require that the simulator be integrated with the actual control stations. If so, then either configuration 4 or configuration 7 are selected, depending on whether the control components can be simulated or must be emulated. Note that the use of actual control hardware (as in configuration 12) is not an option at this point, since it was implicitly rejected by rejecting the staged startup and prerun checkout applications. The hierarchy of application considerations continues in the selection of configurations 3, 1, and 0.

Requirements

Once a configuration has been selected, the data path and functional requirements must be defined. Figure 13 illustrates this procedure. Since a configuration may support more than one application, requirement specification is also dependent on the applications desired. The procedure in figure 13 relates these applications and the selected configuration to the requirement menus of tables VII and VIII. The procedure is entered according to configuration number (ovals at left of fig. 13). The basic requirements are established and, according to the answers to application questions, supplemental requirements are added. Note that figure 13 contains no entry for configuration 0. In this case, there are really no requirements, since the simulations used during facility and control development can be used for configuration 0 applications.

With the requirements established, the integration of the simulator can be costed. Practical considerations for developing these cost estimates are given below.

Costing Considerations

The type of simulator application selected directly impacts the cost of the hardware. The most general configuration,

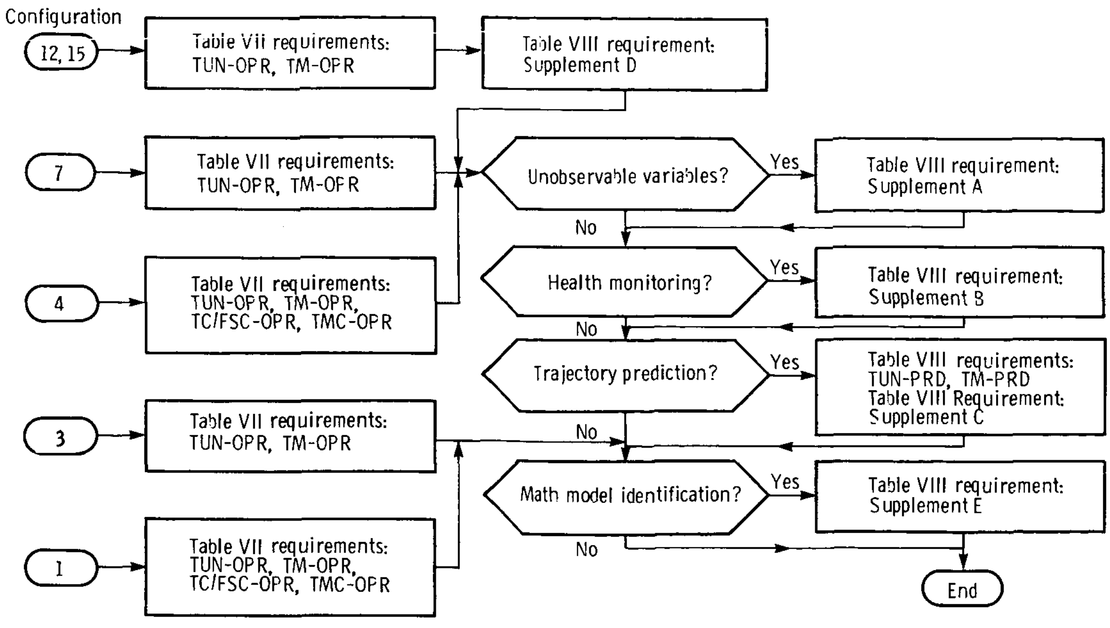


Figure 13.—Requirements specification.

which includes all of the hardware and communication links shown in figure 11, will handle all of the applications discussed in this document. It also carries the highest cost. The cost is also closely related to the required simulation fidelity. For example, the required fidelity of a simulation used for operator training is lower than one used for control system checkout. Higher fidelity implies faster sampling rates, which requires more expensive computers and communications networks. Conversely, cost can be reduced by using a lower fidelity simulation. In the case of the control system checkout application, confidence that all systems function correctly is reduced.

Each of the blocks in figure 11 (except for the sensors and servos) can represent a computer, a system of computers, or a simulation. The lines connecting the blocks represent some sort of communications path. Performance tradeoffs may be used as a means of reducing simulator costs and/or simplifying installation.

Computer Hardware

The most stringent speed requirements of all the elements shown in figure 11 apply to the real-time simulator block. The simulator requires enough computing speed to calculate the system equations of the tunnel and facility subsystems to some specified dynamic accuracy. Therefore, the necessary computer speed is a function of simulation size and/or the highest frequency modes. Typically, minicomputers have been used because of their attractive performance and price ratios for these applications. However, the newer 32-bit supermicrocomputers are becoming increasingly attractive. When several of these microcomputers are combined in a parallel processing network, price and performance ratios meeting or even exceeding that of minicomputers are possible. These parallel processing networks also offer the added benefit of easy expandability to more powerful computing engines. However, appropriate software tools must be provided to allow easy and efficient programming.

Whether a minicomputer or microcomputer or parallel processor should be used depends on the simulator application or combination of applications and the available budget. The parallel processing approach provides the most flexible alternative. The cost can be initially low (with a small number of processors), but can grow as simulation bandwidth increases or as the number of applications grows. This expansibility is an important feature, in light of the difficulty in establishing the necessary simulation fidelity and evaluating computer performance to meet application requirements. Alternatively, if only a low fidelity application (such as operator training) is used, then one of the newer supermicrocomputers alone may be sufficient to accomplish the task.

For several of the configurations, some degree of emulation of the control system is required. There are a number of ways to accomplish this as follows:

(1) The control law equations can be executed on the simulator hardware.

(2) A general purpose microcomputer or network of microcomputers can be used.

(3) The actual control system hardware can be duplicated.

Options 1 and 2 allow lower hardware cost at the expense of additional software development. Although software development costs are difficult to evaluate, as a general rule, these costs will greatly outweigh the cost of the hardware.

The third option provides the highest performance and the most accuracy. It also appears, at first, to be the most expensive. However, there are several advantages which may balance the high initial cost. The first advantage is that the control software can be transported, since the emulated control uses the same hardware as the actual control. Thus, the control algorithms do not have to be recoded to run on the simulator or on a general purpose microcomputer. Secondly, any modifications and/or additions to the control software can be easily and safely tested on the emulated control-simulator system. This testing can occur in parallel with a facility run, if desired. Also, since a FCS usually includes an operators console, replication of the actual operators console would obviously provide the most realistic effect in the operator training application.

Finally, replication of the control system hardware provides built-in control computer redundancy. By incorporating a computer functionality test in the control software, the redundant control computers can take over in the event of a primary computer failure. If a simulator application was using the redundant control computer at that time, the application would be aborted in favor of the failure recovery.

An approach which has not been discussed is the use of the FCS to implement a real-time simulation. If the FCS contains sufficient computing resources, it may be possible to schedule a simulation to execute at some specified rate. However, the simulator would have to compete with the normal FCS functions for computational resources. Communication with these functions would have to occur over the control network. The data rate would be limited since the simulation would again be competing with normal control functions for use of the network. In short, this approach would most likely be suitable only for low-fidelity simulator applications.

Computer Communications

For most of the simulator applications, information must transfer between computer systems. The rate at which the information must be transferred depends on the application. There are four communication paths shown in figure 11 which are required to handle all of the applications discussed in this document. Path A links the simulator with the facility control system. This link sends sensor values to and receives servo commands from the control system in the staged start up application. It can also provide information to the control system for adaptive and predictive control algorithms. The highest data transfer rates will most likely occur on this path. Since path B is an emulation of path A, its data transfer rates will be similar. Paths A1 and B1 allow communication with the facility control station and the emulated facility control station. Since these links will generally be used for operator information data transfers, the data rates will be significantly lower than for paths A and B.

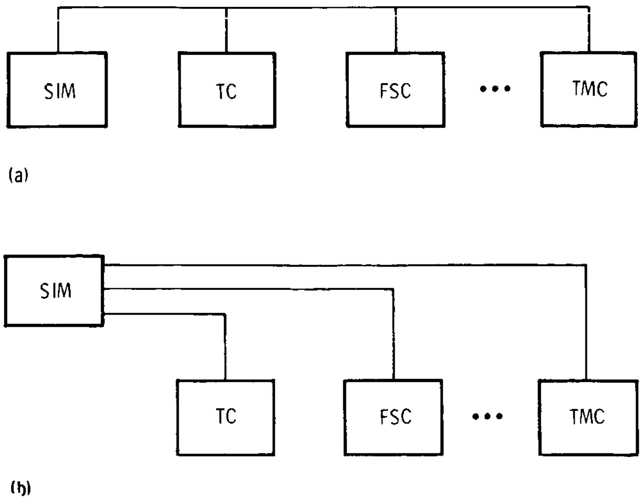
To begin the examination of the requirements for these data paths, it is useful to review the types of communication interfaces available. Table IX lists these interfaces and their relative merits. The first, and least complex, is the analog interface. This consists of converting the digital representations of simulator variables to analog signals, which are transmitted by using standard cabling techniques. Likewise, simulator inputs are read in as analog signals and converted to their digital equivalents. Through the use of amplifiers these signals can be transmitted over long distances. However, the installation and hardware costs are high because of the number of conductors required and the cost of analog to digital and digital to analog converters. The analog interface also suffers from susceptibility to noise. Serious consideration must be given to noise immunity, especially for cables in a high electromagnetic interference (EMI) environment. This would be especially applicable to communications path A in figure 11. Since it is likely that the control system will consist of distributed computers located throughout the facility, the simulator communications links will have to be run through the high EMI facility environment. Thus, high noise immunity is desirable.

Noise immunity and performance can be increased through the use of a digital parallel interface. Unfortunately, the typical length of a parallel interface cable is short. Thus, while a parallel interface may work well for communications path B (computers can be physically adjacent), it is an unrealistic choice for communications path A (computers distributed throughout plant).

A serial interface with coaxial cable in a multidrop configuration offers increased range, simple installation (single cable), and low cost. Performance suffers, however, since all data is now transmitted one bit at a time. The multidrop configuration offers a good choice for a communications interface if the data rates are sufficient for the application. This configuration is a good selection for high EMI environments, and it is in fact the communications mechanism used in many commercial distributed control systems. If even higher noise immunity is required, then a fiber-optic media should be considered rather than coaxial cable. Fiber optics is also the media of choice when higher data rates or longer cable lengths are required.

TABLE IX.—TYPES OF COMPUTER INTERFACES

Interface	Cost	Performance	Installation	Range	Noise immunity
Analog	High	Medium	High	High	Low
Parallel	Medium	High	Medium	Low	Medium
Serial (LAN)	Low	Low	Low	Medium	Medium
Serial (multiple point-to-point)	Medium	High	Medium	Medium	Medium
Serial (fiber optic)	Medium	Medium	Low	High	High



(a) Multidrop arrangement.
(b) Multiple point-to-point serial channels.
Figure 14.—Serial interface arrangements.

Figure 14(a) shows the serial multidrop interface as applied to the simulator-distributed control interface (communication path A in fig. 11). A single coaxial cable connects the simulator to each of the control computers. Access to the media is controlled by a software protocol. This avoids multiple access to the media, but incurs a performance penalty. An increase in performance and simplification of software is achieved by the approach shown in figure 14(b). Multiple serial channels are connected in a point-to-point fashion between the simulator and the control computers. The media can be coaxial cable or fiber optics. The cost of the transmitter-receiver electronics is roughly double that of the multidrop configuration. Installation costs are also higher since now more cabling is involved.

The general discussion of computer communications interfaces is now directed to the specific application of interfacing a real-time simulator to the facility control system. Since the simulator can be a custom designed system, any of the above interface types can be implemented. The type of interface available with the facility control system will depend on the design approach used to implement the control system. If the control system is a custom designed unit, then specifications for the design can include appropriate interfaces to a real-time simulator. The more likely situation, however, is the use of an off-the-shelf distributed control system. In this case, the type of interface will depend on what the control system vendor makes available to the user. There are two types of interfaces typically available—synchronous and asynchronous. Generally, the asynchronous serial interface is used for slow-speed peripheral communications, such as with a CRT or printer. The synchronous interface is used for high-speed intercomputer communications. It is important to realize that in some cases the simulator will be sharing the network with the normal control system functions. This could result in greatly reduced net data rates between the simulator and the control system.

The most flexible interfacing capability comes when the control vendor uses a standard computer bus in the control system design. One example of such a bus is the multibus. The multibus is an industry standard bus, with a large selection of computer interface boards available. The availability of the multibus also provides the capability of doing direct-memory-access (DMA) interfacing to the control computers. The DMA interface is a high-speed parallel interface which is useful for short distance communications paths. This interface would be a possibility for communications path B in figure 11.

Based on the previous discussion, some general conclusions can be made concerning the communications interfaces. Communications path A will most likely be a long length cable routed through a high EMI environment. A high data rate will be required since a high fidelity simulation will be used in conjunction with this path. Therefore, multiple point-to-point digital serial channels using coaxial cable or fiber optics would provide a satisfactory communications link. Since communications path B has the same speed requirements as path A, but will likely be a much shorter length, a parallel or DMA type interface would be called for. Finally, communications paths A1 and B1, requiring lower-speed data rates, would be good applications for single serial channels (multidrop connection).

Sample Hardware and Software Analysis

A typical analysis of real-time simulation requirements is given with the assumption that an off-the-shelf control computer is selected for a particular application. Both hardware and software elements must be considered in this analysis. These elements will be discussed separately.

For hardware elements, the following considerations need to be addressed:

(1) Control system update rate
(2) Control computer hardware specifications
(3) Real-time simulation update rate
(4) Simulation computer hardware specifications
(5) Control system interface method and performance
(6) Is emulated control required?

The control system update rate gives the required time for the control computer to read in data, compute a control algorithm, and output results. Based on the control system update rate and the control computers' hardware specifications, the remaining computing time that can be allocated to support a real-time simulation can be estimated. This time may be sufficient to support the execution of the facility simulation on the control computer itself. The real-time simulation update rate determines if this can be done. The more likely situation, however, is the need for a separate real-time simulation computer. The hardware requirements for the simulation computer will be determined by (1) the simulation code it must execute, (2) the update rate required for real time, and (3) the interface to the control computer. For example, the complexity of the simulation code must be analyzed to determine if it can be executed on some available computer system within the required update interval. This process can be expedited through the use of a representative benchmark code. The benchmark would be typical of the calculations needed for the real-time simulation. If possible, the benchmark could be run on several candidate computer systems. Timing information generated through this process would identify which computer systems meet the real-time simulation requirement. An additional benefit from this procedure would be information regarding the software techniques used to achieve real-time performance. An example of this would be two computer systems which meet the speed specification, but one requires assembly language programming while the other is programmed entirely in a high-level language. The software development costs for the former will be higher than those of the latter.

Besides meeting the performance requirements for the real-time simulation code, the communications between the simulation computer and the control computer must be analyzed. The communications performance requirements can be estimated from the number of variables which must be transferred between the control computer and the simulation computer. Communication must take place within some time period, which is a fraction of the control computers' update time slice. The net communication rate (i.e., megabits per second for serial interfaces or megabytes per second for parallel interfaces) should also take into account any overhead due to software protocols. As an example of a serial communication rate calculation, the number of bits per second which must be transmitted can be calculated as follows:

$$\frac{\text{bits}}{\text{sec}} = \frac{(\text{number of data words})\left(\dfrac{\text{bits}}{\text{word}}\right) + \text{overhead}}{\text{communication period}}$$

where the number of data words is defined by the control algorithm/facility simulation interface and the overhead is defined by the communication protocol. The actual communication method used (serial multidrop, parallel, etc.) may be dictated by the communication interface provided by the control computer. If this interface is inadequate, a custom designed interface may be required at additional cost.

The need for an emulated control should be established based on the selected simulator configuration. If an emulated control is required, then a determination of the implementation must be made. As discussed earlier, the options include full duplication of control hardware, emulation using different hardware or software, or a full software emulation. Budgetary constraints may heavily influence the selection, but caution should be exercised when estimating software costs. In essence, the initially high cost of full-control hardware duplication may pay off in lower long-term software costs.

Finally, the software aspect of real-time simulation development must be examined. For software elements, the following considerations need to be addressed:

(1) Simulation development language and development tools
(2) Software techniques required for real-time performance
(3) Interactive operating system for run-time support
(4) Control computer software interface

The availability of high-level development languages results in significant savings of software development costs. An interactive operating system for run-time support is very critical, especially in the initial simulation development stages. If this system is not commercially available, then considerable time and effort may be expended developing one. As mentioned in the discussion on simulation benchmarking, the software techniques required to achieve real-time performance must also be considered. Again, the required technique (i.e., assembly language programming, system software modification, etc.) may translate into an expensive software development effort. The control computer software must be examined to determine if the simulation computer can be interfaced with reasonable effort. In some cases it may be necessary to contract with the control computer vendor and develop the necessary interface software.

The above considerations must be made within the constraints of budgetary limitations. For example, lower-cost simulation computer hardware and interfacing hardware can be used if a lower fidelity simulation is accepted. Here a reduced number of potential simulator applications (due to a lower fidelity simulation) is traded for lower cost. Thus, it is important to realize that the priority of the real-time simulator and its applications must be established before the hardware and software requirements analysis is done.

Concluding Remarks

Simulation support for facility (e.g., wind tunnel) operations may offer significant improvements in operational costs, reliability, and safety. The actual benefits are highly dependent on the complexity of the operational procedures and/or the control system. An important part of requirement specifications for a facility control and data system should be an evaluation of the benefits and costs associated with simulator support of facility operations.

The applications and requirements presented for the general facility and control system should reduce the time and effort necessary for such an evaluation. The applications that are described impact all operational aspects. These aspects are test plan development, operator training, system validation, facility runs, and results analysis. Upon selection of applications (based on projected operational requirements for the specific facility), a configuration study may be performed to determine the impact of simulator integration into the control system design. The logic of configuration selection presented should aid in this effort. The procedure for establishing functional and interconnection requirements, on the basis of application and configuration selection, is discussed. Once requirements have been established, a cost analysis must be performed to see if projected benefits outweigh simulator integration costs. This procedure may be repeated for various groups of applications until a final set of requirements is formulated.

Lewis Research Center
National Aeronautics and Space Administration
Cleveland, Ohio, September 22, 1986

Bibliography

Bailey Controls: Network 90, Computer Interface Unit, Programmer's Reference Manual. Product Information E93-905, 1984.

———. Network 90, Compute Interface Unit (NCIU02). Product Information E93-905-2, 1984.

Fisher Controls: Type DH6008 Highway Interface Unit. Product Bulletin 4.10:DH6008, Jan. 1983.

———. PRoVOX Instrumentation Systems. Product Bulletin 4:001, Apr. 1983.

Goldberger, A.: A Designer's Review of Data Communications. Computer Design, vol. 20, no. 5, May 1981, pp. 103-112.

Kompass, E.J.: A Long Perspective on Integrated Process Control Systems. Control Eng., vol. 28, no.9, Aug. 1981, pp. 4-9.

Shoch, J.F.; et al.: Evolution of the Ethernet Local Computer Network. Computer, vol. 15, no. 8, Aug. 1982, pp. 10-27.

Westinghouse Distributed Processing Family. Westinghouse Industry Electronics Document IED, 155.

1. Report No. NASA TP-2672	2. Government Accession No.	3. Recipient's Catalog No.
4. Title and Subtitle Applications and Requirements for Real-Time Simulators in Ground-Test Facilities		5. Report Date December 1986
		6. Performing Organization Code 505-40-74
7. Author(s) Dale J. Arpasi and Richard A. Blech		8. Performing Organization Report No. E-3189
		10. Work Unit No.
9. Performing Organization Name and Address National Aeronautics and Space Administration Lewis Research Center Cleveland, Ohio 44135		11. Contract or Grant No.
		13. Type of Report and Period Covered Technical Paper
12. Sponsoring Agency Name and Address National Aeronautics and Space Administration Washington, D.C. 20546		14. Sponsoring Agency Code
15. Supplementary Notes		

16. Abstract

This report relates simulator functions and capabilities to the operation of ground test facilities, in general. The potential benefits of having a simulator are described to aid in the selection of desired applications for a specific facility. Configuration options for integrating a simulator into the facility control system are discussed, and a logical approach to configuration selection based on desired applications is presented. The functional and data path requirements to support selected applications and configurations are defined. Finally, practical considerations for implementation (i.e., available hardware and costs) are discussed.

17. Key Words (Suggested by Author(s))	18. Distribution Statement		
Facilities; Wind Tunnel; Simulators; Computers; Process control; Ground testing	Unclassified – unlimited STAR Category 62		
19. Security Classif. (of this report) Unclassified	20. Security Classif. (of this page) Unclassified	21. No of pages 24	22. Price* A02

*For sale by the National Technical Information Service, Springfield, Virginia 22161

NASA-Langley, 1986